1. (Original) A computer implemented method of exchanging data between software applications comprising:

publishing a list of one or more data fields used by one or more software applications;

mapping the published data fields;

flagging mapped data fields by at least one of the one or more software applications;

matching flagged data fields with a super-schema to define a sub-schema; and

using the sub-schema to validate data files to be exchanged by the one or more software applications.

2. (Original) A computer implemented method of a first software application exchanging data with a second software application comprising:

publishing a list of one or more data fields used by the first software application; wherein the published data fields are mapped to a list of data fields used by the second software application;

flagging a subset of the mapped published data fields; wherein the flagged data fields are matched with a super-schema to define a sub-schema; and

using the sub-schema to validate data files to be exchanged with the second software application.

3. (Original) A computer implemented method of a first software application exchanging data with a second software application comprising:

receiving a list of one or more data fields used by the second software application;

mapping the received data fields to a list of data fields used by the first software application;

receiving a request identifier file flagging a subset of the mapped received data fields;

matching the flagged data fields with a super-schema to define a sub-schema; and

using the sub-schema to validate data files to be exchanged with the second software application.

4. (Original) A computer system for exchanging data between software applications comprising:

a processor for receiving and transmitting data; and

a memory coupled to the processor, the memory having stored therein sequences of instructions which, when executed by the processor, cause the processor to publish a list of one or more data fields used by one or more software applications, map the published data fields to other published data fields, flag a subset of the mapped data fields, match the flagged data fields with a super-schema to define a sub-schema, and use the sub-schema to validate data files to be exchanged by the one or more software applications.

5. (Original) The method as claimed in claim 1 further comprising:

monitoring published data field for flag changes.

6. (Original) The method as claimed in claim 2 further comprising:

monitoring published data field for flag changes.

7. (Original) The method as claimed in claim 3 further comprising:

monitoring published data field for flag changes.

8. (Original) The system as claimed in claim 4 further comprising instructions which, when executed by the processor, cause the processor to:

monitor published data fields for flag changes.

9. (Original) The method as claimed in claim 5 further comprising:

updating the sub-schema in accordance with monitored flag changes of the published data fields.

10. (Original) The method as claimed in claim 6 further comprising:

updating the sub-schema in accordance with monitored flag changes of the published data fields.

11. (Original) The method as claimed in claim 7 further comprising:

updating the sub-schema in accordance with monitored flag changes of the published data fields.

12. (Original) The system as claimed in claim 8 further comprising instructions which, when executed by the processor, cause the processor to:

update the sub-schema in accordance with monitored flag changes of the published data fields.

13. (New) A computer implemented method of exchanging information between first and second software applications comprising:

publishing a first list of first fields of the first software application in a first store by the first software application;

publishing a second list of second fields of the second software application in a second store by the second software application, which is requesting the information from the first software application;

mapping the first and second fields;

flagging a number of mapped fields by the second software application;

matching flagged fields with a super-schema to define a sub-schema by the first software application;

creating a file containing the information from the flagged fields using the sub-schema by the first software application;

using the sub-schema to validate the file by the first software application; and

sending the file to the second software application by the first software application.